Automatic Tuning of Denoising Algorithms Parameters without Ground Truth

Arthur Floquet, Sayantan Dutta, Emmanuel Soubies, Duong-Hung Pham, Denis Kouamé, Denis Kouame

Automatic Tuning of Denoising Algorithms Parameters without Ground Truth

Arthur Floquet, Sayantan Dutta, *Member, IEEE*, Emmanuel Soubies,
Duong-Hung Pham, *Member, IEEE*, and Denis Kouame, *Senior Member, IEEE*

***Abstract*—Denoising is omnipresent in image processing. It is usually addressed with algorithms relying on a set of hyperparameters that control the quality of the recovered image. Manual tuning of those parameters can be a daunting task, which calls for the development of automatic tuning methods. Given a denoising algorithm, the best set of parameters is the one that minimizes the error between denoised and ground-truth images. Clearly, this ideal approach is unrealistic, as the ground-truth images are unknown in practice. In this work, we propose unsupervised cost functions — i.e., that only require the noisy image — that allow us to reach this ideal gold standard performance. Specifically, the proposed approach makes it possible to obtain an average PSNR output within less than 1% of the best achievable PSNR.**

***Index Terms*—Bilevel optimization, Denoising, Hyperparameter tuning.**

I. Introduction

NOISE is inherent to any imaging device. It comes from a variety of sources and is modeled in a variety of ways. When considering additive noise, the corrupted measurements $\mathbf{y} \in \mathbb{R}^N$ follow the model :

$$\mathbf{y} = \mathbf{x} + \mathbf{n}, \tag{1}$$

with $\mathbf{x} \in \mathbb{R}^N$ the clean image we wish to recover, and $\mathbf{n} \in \mathbb{R}^N$ the noise. Many denoising algorithms, denoted $A_{\boldsymbol{\theta}}$, have been proposed to address this task and provide estimates given by $A_{\boldsymbol{\theta}}(\mathbf{y})$, *e.g*, [1]–[3], to cite few. The quality of these estimates depends on the chosen parameters $\boldsymbol{\theta} \in \mathbb{R}^d$. In practice, however, manual tuning of these parameters is far from being trivial, even for low numbers of parameters such as two or three. As such, finding ways to automatically tune these parameters is of major importance and constitutes an active area of research. Most existing approaches use a mapping $\boldsymbol{\Theta}_{\boldsymbol{\lambda}} : \mathbf{y} \mapsto \boldsymbol{\theta}$, itself parameterized by $\boldsymbol{\lambda} \in \mathbb{R}^{d'}$, that maps an image and/or its features (e.g., noise level, noise type, image dynamic, image content) to a set of parameters $\boldsymbol{\theta}$. The best $\boldsymbol{\lambda}$, *i.e.* $\boldsymbol{\lambda}^*$, is found by minimizing the expectancy $\mathbb{E}$ of a discrepancy measure $\mathcal{L}(\cdot,\cdot)$ between the denoised images and ground truth images:

$$\boldsymbol{\lambda}^* := \arg\min_{\boldsymbol{\lambda}\in\mathbb{R}^{d'}} \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})} [\mathcal{L}(A_{\boldsymbol{\Theta}_{\boldsymbol{\lambda}}(\mathbf{y})}(\mathbf{y}), \mathbf{x})]. \tag{2}$$

There are several possibilities to define $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}$ with increasing degree of sophistication:

- $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}$ is a constant mapping (i.e., $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}(\mathbf{y}) = \boldsymbol{\theta} \in \mathbb{R}^d$). Solving (2) then consists in finding a fixed set of parameters $\boldsymbol{\theta}$ such that, on average, the estimates are good, *e.g*, [4].
- $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}$ is defined using features extracted from $\mathbf{y}$. For instance, $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}(\mathbf{y}) = \lambda_1 + \lambda_2 \hat{\sigma}(\mathbf{y})$ where $\hat{\sigma}(\mathbf{y})$ is an estimator of the noise variance. Here, solving (2) allows us to learn a mapping $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}$ that adjusts algorithm parameters according to a *known* and *predefined* model, *e.g*, [5].
- $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}$ is a neural network. In this case, solving (2) allows us to learn a mapping $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}$ that adjusts algorithm parameters according to an *unknown* model, *e.g*, [6], [7]. This greater flexibility comes at the price of a larger number of parameters $\boldsymbol{\lambda}$ to learn.

Typically, all these methods require a dataset and work in a supervised way. Since this is not always feasible, unsupervised alternatives have been developed [8]–[11]. The main idea behind these approaches is to define an unsupervised loss (*i.e.*, that does not depend on $\mathbf{x}$) achieving the same minimizers as the supervised counterpart. Nevertheless, constructing $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}$ and finding $\boldsymbol{\lambda}$ remains challenging.

To discard the need of defining and training a mapping $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}$ on a dataset, one could directly fit $\boldsymbol{\theta}$ on individual images $\mathbf{y}$. The ideal estimate would be:

$$\begin{aligned} &\mathbf{x}^* := A_{\boldsymbol{\theta}^*}(\mathbf{y}) \\ &\text{with } \boldsymbol{\theta}^* := \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^d} \mathcal{C}^*_{\mathbf{y}}(\boldsymbol{\theta}) \ , \ \mathcal{C}^*_{\mathbf{y}}(\boldsymbol{\theta}) := \mathcal{L}(A_{\boldsymbol{\theta}}(\mathbf{y}), \mathbf{x}). \end{aligned} \tag{3}$$

Yet, this formulation is impractical as it requires knowing $\mathbf{x}$ to obtain $\mathbf{x}^*$. In the following, $\mathbf{x}^*$ will be our gold standard, that is, the best estimate we can expect for a given image $\mathbf{y}$ and algorithm $A_{\boldsymbol{\theta}}$. For Gaussian noise, methods such as the famous generalized cross-validation (GCV) [12] and its variants or the Stein's unbiased risk estimate (SURE) [13], which depends only on the noisy data, can be used in place of the true mean-squared error (MSE). The SURE optimization is, however, challenging in the general case and requires the use of approximations [14] or Monte Carlo approaches [15]. It is noteworthy to mention that other metrics that do not require the reference image have also been proposed [16], [17].

Here, inspired by [8]–[11], we propose alternative unsupervised cost functions $\widehat{\mathcal{C}}$ and inference schemes $\widehat{\mathcal{I}}$ such that:

$$\begin{aligned} &\mathbf{x}^* \approx \hat{\mathbf{x}} := \widehat{\mathcal{I}}(\hat{\boldsymbol{\theta}}, \mathbf{y}) \\ &\text{with } \hat{\boldsymbol{\theta}} := \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^d} \widehat{\mathcal{C}}_{\mathbf{y}}(\boldsymbol{\theta}). \end{aligned} \tag{4}$$

Let us emphasize that the inference scheme in (4) is not directly $A_{\hat{\boldsymbol{\theta}}}(\mathbf{y})$. Indeed, as detailed in Section II, the proposed unsupervised cost functions may require adapting the inference

This work was supported by the CIMI Excellence Laboratory, ANR grant ANR-11-LABX-0040 within the French State Programme "Investissements d'Avenir"

A. Floquet, E. Soubies, D.-H. Pham, and D. Kouamé are with the IRIT Laboratory, Université de Toulouse, and CNRS, Toulouse 31400, France. Email: firstname.lastname@irit.fr. (*Corresponding author : Arthur Floquet*)

S. Dutta is with the Department of Radiology, Weill Cornell Medicine, New York, NY 10022, USA. Email: sdu4004@med.cornell.edu.

scheme. As such, we will systematically specify both the cost function and the inference scheme.

Remark 1: The use of the cost function $\mathcal{C}^*$ could be avoided in (3) and replaced with $\mathcal{L}$ directly. We write (3) this way to be consistent with the following unsupervised cost functions $\widehat{\mathcal{C}}$, which don't reduce to simple discrepancy measures between two images. Moreover, we will use $\cdot^*$ and $\widehat{\cdot}$ for supervised and unsupervised objects, respectively.

Paper Outline

The paper is structured as follows. The proposed method is detailed in Section II. Then, in Section III, we deploy our approach to tune the parameters of a recently published denoiser [3] that can handle a variety of noise types. Finally, discussions and conclusions are provided in Section IV.

II. Method

Let us start this section by restating the key difference between our approach in Eq. (4) and the more standard one in Eq. (2) (with its unsupervised counterparts [8]–[11]). Notably, there is no training phase in (4). We don't need to create and train a mapping $\boldsymbol{\Theta}_{\boldsymbol{\lambda}}$ over a dataset that will then be used to infer estimates on new data. Instead, our inference is done by solving (4) directly for individual images $\mathbf{y}$.

We also emphasize that our aim is not to build a new denoiser, but an automatic way to select hyperparameters of a given denoiser, using only the input noisy image. For illustration purposes, we use the DeQuIP algorithm [3] described in Section II-C.

A. The Cost Functions and Inference Schemes Definition

We drew our inspiration from existing unsupervised learning methods. Instead of training a denoising neural network f of parameters ω, say $f_{\boldsymbol{\omega}}$, on noisy-clean pairs $(\mathbf{y}, \mathbf{x})$, Noise2Noise (N2N) [8] proposes to train it on pairs of noisy images $(\mathbf{y}, \mathbf{y}')$ with $\mathbf{y}$ and $\mathbf{y}'$ two noisy versions of the same clean image $\mathbf{x}$. Lehtinen *et al.* [8] showed that for several noise types, $\mathcal{L}$ can be chosen accordingly so that:

$$\arg\min_{\boldsymbol{\omega}} \mathop{\mathbb{E}}_{(\mathbf{y},\mathbf{y}')} \mathcal{L}(f_{\boldsymbol{\omega}}(\mathbf{y}), \mathbf{y}') = \arg\min_{\boldsymbol{\omega}} \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})} \mathcal{L}(f_{\boldsymbol{\omega}}(\mathbf{y}), \mathbf{x}). \quad (5)$$

For example, if $\mathbf{y}, \mathbf{y}'$ are two corrupted versions of $\mathbf{x}$ with independent additive zero-mean noise (i.e., $\mathbf{y} = \mathbf{x} + \mathbf{n}$, $\mathbf{y}' = \mathbf{x} + \mathbf{n}'$ with $\mathbb{E}[\mathbf{n}] = \mathbb{E}[\mathbf{n}'] = \mathbf{0}$), letting $\mathcal{L}$ be the quadratic error leads to (5):

$$\begin{aligned}
\hat{\boldsymbol{\omega}} &:= \arg\min_{\boldsymbol{\omega}} \mathop{\mathbb{E}}_{(\mathbf{y},\mathbf{y}')} \|f_{\boldsymbol{\omega}}(\mathbf{y}) - \mathbf{y}'\|_2^2 \\
&= \arg\min_{\boldsymbol{\omega}} \mathop{\mathbb{E}}_{(\mathbf{y},\mathbf{x},\mathbf{n}')} \|f_{\boldsymbol{\omega}}(\mathbf{y}) - \mathbf{x} - \mathbf{n}'\|_2^2 \\
&= \arg\min_{\boldsymbol{\omega}} \mathop{\mathbb{E}}_{(\mathbf{y},\mathbf{x},\mathbf{n}')} \left[\|f_{\boldsymbol{\omega}}(\mathbf{y}) - \mathbf{x}\|_2^2 - 2\left\langle f_{\boldsymbol{\omega}}(\mathbf{y}) - \mathbf{x}, \mathbf{n}'\right\rangle\right] \\
&= \arg\min_{\boldsymbol{\omega}} \mathop{\mathbb{E}}_{(\mathbf{y},\mathbf{x})} \|f_{\boldsymbol{\omega}}(\mathbf{y}) - \mathbf{x}\|_2^2 := \boldsymbol{\omega}^*, \qquad (6)
\end{aligned}$$

where the expectation of the dot product cancels as $\mathbb{E}[\mathbf{n}'] = \mathbf{0}$, and $\mathbf{n}$ and $\mathbf{n}'$ (i.e., $\mathbf{y}$ and $\mathbf{n}'$) are independent.

In practice, training is performed using a finite dataset $\Omega_{I,J} := \{\mathbf{y}_i^j = \mathbf{x}_i + \mathbf{n}_i^j\}_{i,j=(1,1)}^{I,J}$. This means that for a noiseless image $\mathbf{x}_i$, J noisy versions are available, with additive noises $\mathbf{n}_i^j$. As such, the true loss in (5) is replaced by an empirical one. This results in a solution $\hat{\boldsymbol{\omega}} \approx \boldsymbol{\omega}^*$ whose quality depends on I, J:

- When both I and J are large, the empirical loss yields a good approximation of the true one.
- When I is small (e.g., $I = 1$), but J is large, we can expect $f_{\widehat{\boldsymbol{\omega}}}$ to perform well on the training set, but not to generalize well. It will effectively learn to output the mean of the $\mathbf{y}_1^j$, which is $\mathbf{x}_1$.
- When J is small (e.g., $J = 2$, with different $\mathbf{n}_i^1$, $\mathbf{n}_i^2$ for each i), $f_{\widehat{\boldsymbol{\omega}}}$ could overfit the dataset $\Omega_{I,J}$ to the point that we systematically obtain $f_{\widehat{\boldsymbol{\omega}}}(\mathbf{y}_i^1) = \mathbf{y}_i^2$, $f_{\widehat{\boldsymbol{\omega}}}(\mathbf{y}_i^2) = \mathbf{y}_i^1$, $\forall i$. Yet, a large I (e.g., $I = 1000$) seems to be sufficient in preventing overfitting as demonstrated by Lehtinen *et al,* [8]. Other factors can be exploited to avoid overfitting when limited data are available. These include network architecture, image type, noise type, or a low number of parameters relative to image size, $\mathrm{card}(\boldsymbol{\omega})/P$, with P the number of image pixels [18]. For instance, it has been shown that U-Net-like networks have a hard time recreating noise [19].
- When both I and J are small (e.g., $I = 1$, $J = 2$), if not carefully designed, $f_{\boldsymbol{\omega}}$ would simply learn to map $\mathbf{y}_1^1$ to $\mathbf{y}_1^2$, and vice-versa.

The proposed method, i.e., fitting algorithm parameters to a single image, corresponds to the last extreme case. We argue that if, instead of a neural network (which can recreate noise), we use a denoising algorithm $A_{\boldsymbol{\theta}}$ with low $\mathrm{card}(\boldsymbol{\theta})/P$, we can define:

$$\widehat{\mathcal{C}}_{\mathbf{y},\mathbf{y}'}^{\mathrm{N2N}}(\boldsymbol{\theta}) := \|A_{\boldsymbol{\theta}}(\mathbf{y}) - \mathbf{y}'\|_2^2, \hat{\mathbf{x}}^{\mathrm{N2N}} := A_{\hat{\boldsymbol{\theta}}}(\mathbf{y}), \quad (7)$$

and have $\hat{\mathbf{x}}^{\mathrm{N2N}} \approx \mathbf{x}^*$ (we remind the reader that here, as for the rest of the paper, $\hat{\boldsymbol{\theta}} := \arg\min \widehat{\mathcal{C}}_{\mathbf{y}}(\boldsymbol{\theta})$). This claim, consistent with (6), is supported by our numerical experiments in Section III where we consistently obtain an estimate $\hat{\mathbf{x}}^{\mathrm{N2N}}$ of the same quality as the gold standard $\mathbf{x}^*$.

Remark 2: The cost $\widehat{\mathcal{C}}_{\mathbf{y},\mathbf{y}'}^{\mathrm{N2N}}$ is presented using the L_2 norm. Yet, other distances $\mathcal{L}$ can be considered, for instance, to handle other types of noise as in [8].

Still, N2N requires two independent noisy versions of the same clean image, which is not very standard in practice. Thus, we explored alternative ideas that extend the N2N one such that a single noise realization $\mathbf{y}$ per clean image $\mathbf{x}$ is required. They revolve around strategies of renoising $\mathbf{y}$ into $\mathbf{z}$ so as to create pairs of noisy images. These ideas and the corresponding costs and inferences we propose in our context are described below.

Noisy-as-Clean (NaC) [9]: The underlying idea is to create a doubly noisy image $\mathbf{z} = \mathbf{y} + \mathbf{n}_{\mathrm{s}}$, with $\mathbf{n}_{\mathrm{s}}$ being simulated noise drawn from the same distribution as $\mathbf{n}$ according to (1), and learn to go from $\mathbf{z}$ to $\mathbf{y}$. Inference is then done on regular noisy images: $\hat{\mathbf{x}} = f_{\widehat{\boldsymbol{\omega}}}(\mathbf{y})$. From the continuity of $\mathcal{L}$, f_{ω}, $p_{\mathbf{y}|\mathbf{x}}$, $p_{\mathbf{z}|\mathbf{y}}$ and assuming that the noise is low, the authors in [9] argue that $\widehat{\boldsymbol{\omega}} \approx \boldsymbol{\omega}^*$. Since f_{ω} is continuous, this implies $\hat{\mathbf{x}} \approx \mathbf{x}^*$. Building upon this idea, we propose the following scheme:

$$\widehat{\mathcal{C}}_{\mathbf{y}}^{\mathrm{NaC}}(\boldsymbol{\theta}) := \mathop{\mathbb{E}}_{\mathbf{z}} \|A_{\boldsymbol{\theta}}(\mathbf{z}) - \mathbf{y}\|_2^2, \hat{\mathbf{x}}^{\mathrm{NaC}} := A_{\hat{\boldsymbol{\theta}}}(\mathbf{y}). \quad (8)$$

Noisier2Noise (Nr2N) [10]: As in NaC, it also proceeds by creating a doubly noisy image $\mathbf{z} = \mathbf{y} + \mathbf{n}_s$, with $\mathbf{n}_s$ following the same distribution as $\mathbf{n}$ according to (1), and learn to go from $\mathbf{z}$ to $\mathbf{y}$. However, it does not require any other assumptions than the sole additive noise. In particular, it is not restricted to a low noise level. As such, $\widehat{\boldsymbol{\omega}} \not\approx \boldsymbol{\omega}^*$, so inference cannot be performed on $\mathbf{y}$. It is instead done on $\mathbf{z}$ and includes a correction step: $\hat{\mathbf{x}} = 2f_{\widehat{\boldsymbol{\omega}}}(\mathbf{z}) - \mathbf{z} = \mathbb{E}[\mathbf{x}|\mathbf{z}]$. This amounts to supervised denoising of the image $\mathbf{z}$, which has a higher noise level than $\mathbf{y}$. To mitigate the effect of this artificially high noise level, one solution is to lower the variance of $\mathbf{n}_s$. The correction step will then depend on the new variance. For Gaussian noise, with $\mathbf{n} \sim \mathcal{N}(0, \sigma)$, NaC uses $\mathbf{n}_s \sim \mathcal{N}(0, \alpha\sigma)$, with $\alpha \in]0, 1]$. The inference becomes $\hat{\mathbf{x}} = \frac{(1+\alpha^2)f_{\widehat{\boldsymbol{\omega}}}(\mathbf{z})-\mathbf{z}}{\alpha^2}$. A tradeoff has then to be made between a lower simulated noise, and an amplification of denoising errors due to the α^2 division. Exploiting the Nr2N idea in our context leads to

$$\begin{aligned} \widehat{\mathcal{C}}^{\mathrm{Nr2N}}_{\mathbf{y}}(\boldsymbol{\theta}) &:= \mathop{\mathbb{E}}_{\mathbf{z}} \|A_{\boldsymbol{\theta}}(\mathbf{z}) - \mathbf{y}\|_2^2 \\ \hat{\mathbf{x}}^{\mathrm{Nr2N}} &:= \frac{(1+\alpha^2)A_{\hat{\boldsymbol{\theta}}}(\mathbf{z}) - \mathbf{z}}{\alpha^2}. \end{aligned} \tag{9}$$

Recorrupted-to-Recorrupted (R2R) [11]: Here, the authors propose to create two doubly noisy images $\mathbf{z}_1 = \mathbf{y} + \mathbf{D}^{\mathrm{T}}\mathbf{n}_{\mathrm{s}}$ and $\mathbf{z}_2 = \mathbf{y} - \mathbf{D}^{-1}\mathbf{n}_{\mathrm{s}}$, with $\mathbf{D}$ being any invertible matrix and $\mathbf{n}_{\mathrm{s}}$ drawn from the same distribution as $\mathbf{n}$, according to (1). Note that for R2R, $\mathbf{n}$ is assumed to be Gaussian. This way, we can write $\mathbf{z}_1 = \mathbf{x} + \mathbf{n}_1$, $\mathbf{z}_2 = \mathbf{x} + \mathbf{n}_2$, with $\mathbf{n}_1$ and $\mathbf{n}_2$ being zero-mean, independent noise vectors. Training can then be done as in N2N. Yet, as in NaC and Nr2N, such unsupervised training can be seen as supervised training with a higher noise level. To deal with this, R2R uses a Monte-Carlo scheme in inference to limit the effect of $\mathbf{n}_s$: $\hat{\mathbf{x}} = \frac{1}{M}\sum_m f_{\widehat{\boldsymbol{\omega}}}(\mathbf{z}_1^m)$ where the superscript m simply represents the m^{th} drawing of $\mathbf{n}_s$. For our fitting problem (4), we propose the scheme:

$$\begin{aligned} \widehat{\mathcal{C}}^{\mathrm{R2R}}_{\mathbf{y}}(\boldsymbol{\theta}) &:= \mathop{\mathbb{E}}_{(\mathbf{z}_1,\mathbf{z}_2)} \|A_{\boldsymbol{\theta}}(\mathbf{z}_1) - \mathbf{z}_2\|_2^2 \\ \hat{\mathbf{x}}^{\mathrm{R2R}} &:= \frac{1}{M}\sum_{m=1}^{M} A_{\hat{\boldsymbol{\theta}}}(\mathbf{z}_1^m). \end{aligned} \tag{10}$$

B. Optimization Aspects

To solve Problem (4), we exploit automatic differentiation to perform gradient descent steps. This implies that $A_{\boldsymbol{\theta}}$ must be differentiable with respect to $\boldsymbol{\theta}$, or at least subdifferentiable. For instance, as explained in Section II-C, hard thresholds need to be replaced with soft thresholds. For $\widehat{\mathcal{C}}^{\mathrm{NaC}}$, $\widehat{\mathcal{C}}^{\mathrm{Nr2N}}$ and $\widehat{\mathcal{C}}^{\mathrm{R2R}}$, we approximate $\mathbb{E}_{\mathbf{z}}$ by drawing a new random $\mathbf{z}^k$ at each iteration $k \in [\![1; K]\!]$. For initialization, we use a fixed $\boldsymbol{\theta}_0$, found by manually tuning $\boldsymbol{\theta}$ for *a single image*. [1]

C. Use case: Denoising via Quantum Interactive Patches

We illustrate our method, without loss of generality, to a denoiser, denoted DeQuIP (*Denoising via Quantum Interactive Patches*) [3] which can handle various types of noise. It is based on two main concepts of quantum mechanics: (i) exploits the Schrödinger equation of quantum physics to construct an adaptive basis, which enables DeQuIP to deal with various noise models; (ii) treats the image as patches, and formalizes the self-similarity between neighbor patches through a term akin to quantum many-body interaction to efficiently preserve the local structures. Each patch behaves as a single-particle system, which interacts with other neighboring patches, thus the whole image acts as a many-body system. Under a potential $\mathbf{V}$, represented by the image pixel values [20], the adaptive basis vector $\boldsymbol{\psi}$ describes the characteristics of a virtual quantum particle with energy E, and satisfies the non-relativistic stationary Schrödinger equation, written as:

$$-\frac{\hbar^2}{2m}\nabla^2\boldsymbol{\psi} = -\mathbf{V}\boldsymbol{\psi} + E\boldsymbol{\psi}, \tag{11}$$

with $\hbar$ being Planck's constant, m the particle mass, and ∇^2 the Laplacian operator. The image-dependent basis is constructed from the wave solutions $\boldsymbol{\psi}_{\boldsymbol{i}}$ of eq. (11) by plugging the noisy image $\mathbf{y}$ as the potential $\mathbf{V}$ of the system. These wave solutions are oscillating functions with a local frequency proportional to $\sqrt{(E-\mathbf{V})/(\hbar^2/2m)}$, thus the frequency is locally adapted to the image pixels' values. The exact behavior of these basis vectors with respect to the potential is determined by the constant $\frac{\hbar^2}{2m}$, which is a parameter of DeQuIP. For a more detailed illustration of these oscillating wave vectors, we refer readers to [20], [21]. Finally, for denoising, the noisy image $\mathbf{y}$ is projected onto this adaptive basis, and the low-value coefficients are thresholded using a soft threshold:

$$\Upsilon_{c_1,c_2}(x) = \begin{cases} 0 & \text{if } |\alpha| \le c_1 \\ \frac{c_2}{c_2-c_1}x - \frac{c_1c_2}{c_2-c_1} & \text{if } -c_2 \le x \le -c_1 \\ \frac{c_2}{c_2-c_1}x + \frac{c_1c_2}{c_2-c_1} & \text{if } c_2 \ge x \ge c_1 \\ 1 & \text{elsewhere} \end{cases}$$

The second key idea of DeQuIP is the integration of quantum interaction theory to consider the image as overlapping patches, where each patch is a single-particle system that interacts with other patches [3]. The interaction between two patches $\mathbf{A}$ and $\mathbf{B}$ is defined as: $\mathbf{I}_{AB} = p\frac{|\mathbf{A}-\mathbf{B}|}{d(A,B)^2}$ with p a parameter controlling the strength of the interaction term, and d the euclidean pixel distances between the center of $\mathbf{A}$ and $\mathbf{B}$. This term is designed to promote local self-similarity, based on the hypothesis that neighboring patches in an image are likely to be more similar than distant ones.

Therefore, DeQuIP has 4 parameters $\boldsymbol{\theta} = \{-\frac{\hbar^2}{2m}, p, c_1, c_2\}$ shared across patches.

III. Experimental Results

A. Material & Data

Implementation is done on *PyTorch* 1.12.0. We used 180×180 clean grayscale photographic images, extracted from the *BSD400* dataset, as ground truth.

B. Comparison of Denoised Images

The estimates $\hat{\mathbf{x}}^{\mathrm{N2N}}$, $\hat{\mathbf{x}}^{\mathrm{NaC}}$, $\hat{\mathbf{x}}^{\mathrm{Nr2N}}$, $\hat{\mathbf{x}}^{\mathrm{R2R}}$ and the gold standard $\mathbf{x}^*$ are compared in the zero-mean Gaussian noise case since it is the only case covered by the four proposed schemes.

[1] Codes and supplementary materials can be found here

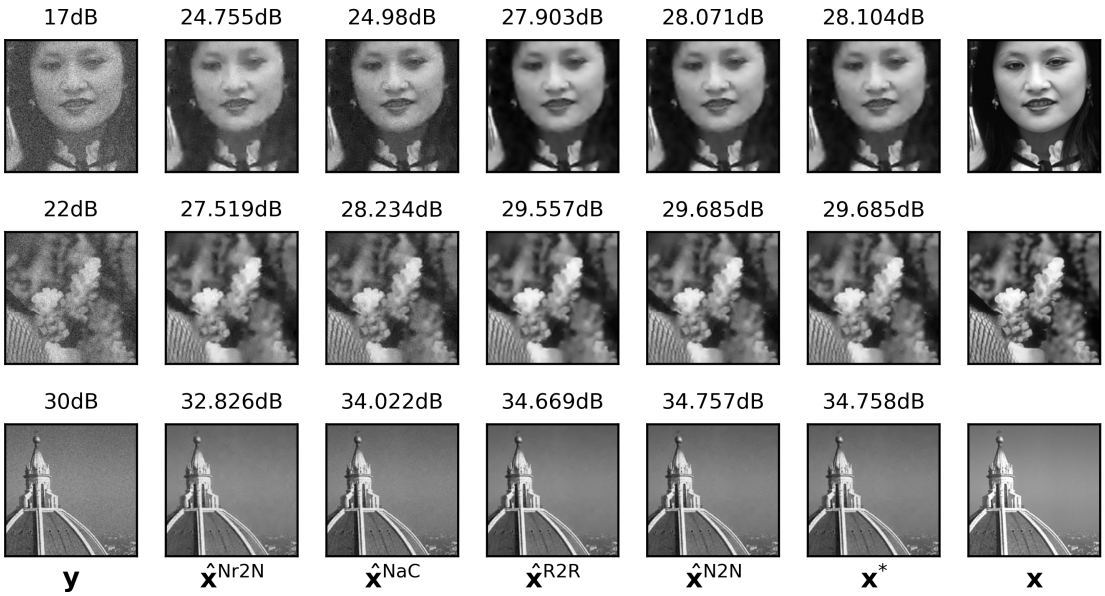

Fig. 1: Exemple of denoising results. The upper scripts of $\hat{\mathbf{x}}$ refer to the corresponding method; $\mathbf{x}^*$ and $\mathbf{x}$ denote respectively the result for the supervised (gold standard) approach and the ground-truth.

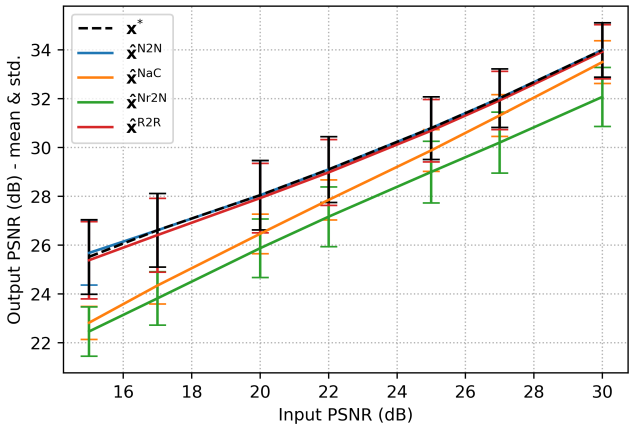

Fig. 2: Comparison of the proposed strategies with the gold standard $\mathbf{x}^*$. The average performance and standard deviation is computed over 35 test images.

We report on Fig. 2 the average performance (measured by output PSNR) and standard deviation over 35 test images as a function of the input PSNR. Examples of noisy and denoised images are presented in Fig. 1. We see, both in metric and visually, that $\hat{\mathbf{x}}^{\mathrm{N2N}}$ and $\hat{\mathbf{x}}^{\mathrm{R2R}}$ achieve the gold standard performance $\mathbf{x}^*$. As expected given the assumptions it relies on, $\hat{\mathbf{x}}^{\mathrm{NaC}}$ yields better results when noise is low enough (i.e., high PSNR), but still underperforms. Finally, $\hat{\mathbf{x}}^{\mathrm{Nr2N}}$ does not perform as well as the other unsupervised methods.

C. Poisson Noise Denoising

As DeQuIP can deal with various noise models, we present in Fig. 3 Poisson noise denoising results, where we can observe that $\hat{\mathbf{x}}^{\mathrm{N2N}}$ is similar to $\mathbf{x}^*$. We used $\widehat{\mathcal{C}}^{\mathrm{N2N}}$ because of its performances, and its capacity to handle Poisson noise.

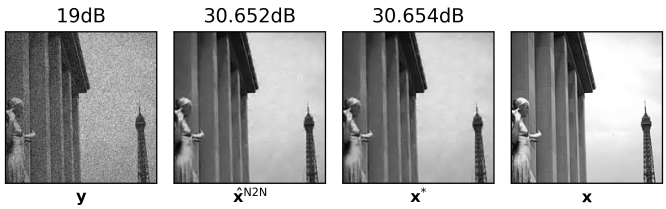

Fig. 3: Poisson noise denoising.

IV. Conclusion

We have proposed a method for the automatic tuning of denoising algorithm parameters, leveraging only the noisy measurements targeted for enhancement. Specially, we introduced several cost functions and inference schemes, two of which yielded results comparable to those obtained with ground truth-tuned parameters.

However, our method comes with certain limitations. The first one arises when only a single noisy image is available, requiring the noise to follow a zero-mean Gaussian distribution. For other types of noise, two independent noisy versions are needed. The second limitation is the need to resolve an optimization problem that involves differentiating the denoising algorithm with respect to its parameters. Although this can be achieved using a simple backpropagation, it can be computationally expensive.

Moving forward, there are potential avenues for exploration. For algorithms A that cannot replicate the identity function, exploring unsupervised losses inspired by Noise2Self [22] and [23] could be fruitful, given their advantage of using a single noise realization $\mathbf{y}$ without renoising it. Another idea worth exploring is the extension of our method to other inverse problems such as image deblurring or super-resolution, possibly building on [24], [25].

REFERENCES

[1] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[2] I. Selesnick, "Total variation denoising via the moreau envelope," *IEEE Signal Processing Letters*, vol. 24, no. 2, pp. 216–220, 2017.

[3] S. Dutta, A. Basarab, B. Georgeot, and D. Kouamé, "A novel image denoising algorithm using concepts of quantum many-body theory," *Signal Processing*, vol. 201, p. 108 690, Dec. 2022.

[4] L. Calatroni *et al.*, "Bilevel approaches for learning of variational imaging models," *Variational Methods: In Imaging and Geometric Control*, vol. 18, no. 252, p. 2, 2017.

[5] P. Nguyen, E. Soubies, and C. Chaux, "MAP-informed unrolled algorithms for hyper-parameter estimation," in *International Conference on Image Processing*, IEEE, Sep. 2023.

[6] B. M. Afkham, J. Chung, and M. Chung, "Learning regularization parameters of inverse problems via deep neural networks," *Inverse Problems*, vol. 37, no. 10, p. 105 017, 2021.

[7] A. Kofler *et al.*, "Learning regularization parameter-maps for variational image reconstruction using deep neural networks and algorithm unrolling," *arXiv Preprint*, Jan. 2023.

[8] J. Lehtinen *et al.*, "Noise2Noise: Learning image restoration without clean data," in *International Conference on Machine Learning*, PMLR, Mar. 2018.

[9] J. Xu *et al.*, "Noisy-as-Clean: Learning self-supervised denoising from corrupted image," *IEEE Transactions on Image Processing*, vol. 29, pp. 9316–9329, 2020.

[10] N. Moran, D. Schmidt, Y. Zhong, and P. Coady, "Noisier2Noise: Learning to denoise from unpaired noisy data," in *Conference on Computer Vision and Pattern Recognition*, IEEE/CVF, Oct. 2019.

[11] T. Pang, H. Zheng, Y. Quan, and H. Ji, "Recorrupted-to-Recorrupted: Unsupervised deep learning for image denoising," in *Conference on Computer Vision and Pattern Recognition*, IEEE/CVF, 2021, pp. 2043–2052.

[12] G. H, Golub, H. M, and W. G, "Generalized cross-validation as a method for choosing a good ridge parameter," en, *Technometrics,*, vol. 21, no. 12, May 1979.

[13] C. M. Stein, "Estimation of the mean of a multivariate normal distribution," *The annals of Statistics*, pp. 1135–1151, 1981.

[14] C. Deledalle, S. Vaiter, J. Fadili, and G. Peyré, "Stein unbiased gradient estimator of the risk (sugar) for multiple parameter selection," *SIAM Journal on Imaging Sciences*, vol. 7, no. 2, pp. 2448–2487, 2014.

[15] S. Ramani, T. Blu, and M. Unser, "Monte-Carlo Sure: A Black-Box Optimization of Regularization Parameters for General Denoising Algorithms," en, *IEEE Transactions on Image Processing*, vol. 17, no. 9, pp. 1540–1554, Sep. 2008.

[16] Xiang Zhu and P. Milanfar, "Automatic Parameter Selection for Denoising Algorithms Using a No-Reference Measure of Image Content," en, *IEEE Transactions on Image Processing*, vol. 19, no. 12, pp. 3116–3132, Dec. 2010.

[17] X. Kong and Q. Yang, "No-reference image quality assessment for image auto-denoising," *International Journal of Computer Vision*, vol. 126, no. 5, pp. 537–549, May 2018.

[18] X. Ying, "An Overview of Overfitting and its Solutions," en, *Journal of Physics: Conference Series*, vol. 1168, p. 022 022, Feb. 2019.

[19] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep Image Prior," in *Conference on Computer Vision and Pattern Recognition*, IEEE/CVF, 2018, pp. 1867–1888.

[20] S. Dutta, A. Basarab, B. Georgeot, and D. Kouamé, "Quantum mechanics-based signal and image representation: Application to denoising," *IEEE Open Journal of Signal Processing*, vol. 2, pp. 190–206, 2021.

[21] S. Dutta, A. Basarab, B. Georgeot, and D. Kouamé, "Image denoising inspired by quantum many-body physics," in *2021 IEEE International Conference on Image Processing (ICIP)*, 2021, pp. 1619–1623.

[22] A. Krull, T.-O. Buchholz, and F. Jug, "Noise2void-learning denoising from single noisy images," *Conference on Computer Vision and Pattern Recognition*, pp. 2129–2137, 2019.

[23] J. Batson and L. Royer, "Noise2self: Blind denoising by self-supervision," *International Conference on Machine Learning*, pp. 524–533, 2019.

[24] Z. Xia and A. Chakrabarti, "Training image estimators without image ground truth," in *Advances in Neural Information Processing Systems*, H. Wallach *et al.*, Eds., vol. 32, Curran Associates, Inc., 2019.

[25] J. Tachella, D. Chen, and M. Davies, "Unsupervised learning from incomplete measurements for inverse problems," in *Proceedings of the 36th Conference on Neural Information Processing Systems*, 2022.